

Motion Vector Estimation Using Line-Square Search Block Matching Algorithm for Video Sequences

Ding Gui-guang

*Broadband Networks & digital Media Lab, Automation Department, Tsinghua University, Beijing 100084, China
Email: dinggg@mail.tsinghua.edu.cn*

Guo Bao-long

*School of Electromechanical Engineering, Xidian University, Xi'an 710071, China
Email: blguo@xidian.edu.cn*

Received 19 May 2003; Revised 16 December 2003; Recommended for Publication by Reha Civanlar

Motion estimation and compensation techniques are widely used for video coding applications but the real-time motion estimation is not easily achieved due to its enormous computations. In this paper, a new fast motion estimation algorithm based on line search is presented, in which computation complexity is greatly reduced by using the line search strategy and a parallel search pattern. Moreover, the accurate search is achieved because the small square search pattern is used. It has a best-case scenario of only 9 search points, which is 4 search points less than the diamond search algorithm. Simulation results show that, compared with the previous techniques, the LSPS algorithm significantly reduces the computational requirements for finding motion vectors, and also produces close performance in terms of motion compensation errors.

Keywords and phrases: motion estimation, diamond search algorithm, video coding.

1. INTRODUCTION

In video coding, the high correlation between successive frames can be exploited to improve coding efficiency, which is usually achieved by using motion estimation (ME) and motion compensation technology. Many ME methods have been studied in an effort to reduce the computational complexity of the ME, such as block matching algorithms (BMA), parametric/motion models, optical flow, and pel-recursive techniques. Among these methods, BMA seems to be the most popular method due to its effectiveness and simplicity for both software and hardware implementations. BMA is also widely adopted by various video coding standards, such as MPEG-1 [1], MPEG-2 [2], MPEG-4 [3], H.261 [4], H.263 [5], and H.26L [6].

It is obvious that in using BMA, the simplest and most accurate strategy is the full search algorithm (FS), sometimes referred to as the exhaustive search of the brute-force search. The FS algorithm gives an optimum solution, in terms of prediction quality, by exhaustively searching over all possible blocks within the search window. However, the computational complexity of FS has motivated a host of suboptimal but faster search strategies. Thus, efficient algorithms such as the three-step search algorithm (TSS) [7], the four-step search algorithm (FSS) [8], the two-dimensional logarithmic search algorithm (TDL) [9], the cross search algo-

rithm (CS) [10], and the diamond search algorithm (DS) [11] have been developed to reduce the computational complexity. Among the proposed fast algorithms, the DS algorithm recommended by verify model (VM) of MPEG-4 [12] became the most widely used technique because of its simplicity and effectiveness [13]. According to the characteristics of the center-biased motion vector distribution, the DS algorithm employs two search patterns as illustrated in Figure 1. The first pattern, called large diamond search pattern (LDSP), comprises nine checking points of which eight points surround the center one to compose a diamond shape, while the second pattern consisting of five checking points forms a small diamond shape, called small diamond search pattern (SDSP). In the searching procedure of the DS algorithm, LDSP is repeatedly used until the step in which the minimum block distortion (MBD) occurs at the center point. The search pattern is then switched from LDSP to SDSP to reach the final search stage. Among the five checking points in SDSP, the position yielding the MBD provides the motion vector of the best matching point.

There are two main drawbacks in the DS algorithm. Firstly, for stationary blocks, only 5 search points are needed in ideal condition. Nevertheless, 13 search points are needed in the DS algorithm, that is to say the DS algorithm needs to be improved when it searches stationary blocks. Secondly, the DS algorithm does not employ the spacial directionality

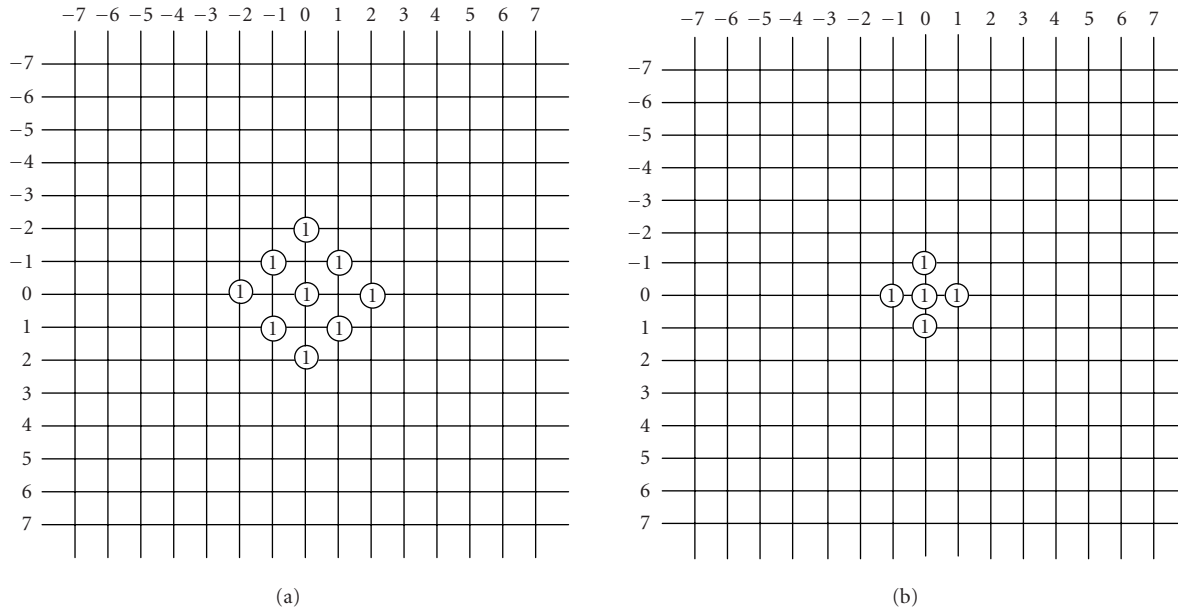


FIGURE 1: Two search patterns are employed in the DS algorithm: (a) LDSP and (b) SDSP.

of the sum of absolute difference (SAD) distribution to determine the area at which the best matching point is, which results in a mass of search redundancy.

According to the above questions, this paper presents a novel line-square parallel search (LSPS) algorithm for sub-optimal block ME. The number of search points could be reduced by introducing the parallel processing idea and line search strategy. Simulation results show that the LSPS algorithm is not only reducing computational complexity but also improving the quality performance when compared to the DS algorithm.

The organization of this paper is as follows. Section 2 describes the basic properties of the block motion fields and the SAD distribution, and then explains how to use these properties in the LSPS algorithm. A performance analysis is also carried out to demonstrate why the LSPS algorithm is fast and effective. Section 3 presents some simulation results of the LSPS algorithm in comparison with FS, TSS, FSS, and DS. Finally, the conclusions are given in Section 4.

2. THE LINE-SQUARE PARALLEL SEARCH ALGORITHM

2.1. Basic properties

Generally speaking, the search pattern and search strategy employed in the fast algorithm jointly determine not only its search speed but also the resulting performance. In order to design the best optimal pattern and strategy, it is very important to investigate the basic properties of the block motion fields and the SAD distribution. Following are some of the basic properties of the BMA when applied to typical video sequences.

Property 1. The distribution of the block motion fields is center-biased. This means that smaller displacements are more probable and the motion vector (0,0) has the highest probability of occurrence. Namely, most blocks are stationary or quasi-stationary [13]. This is depicted in Figure 2a, in which more than 80% of the blocks are stationary or quasi-stationary (within a central 3×3 area) in the Foreman sequence.

Property 2. The error surface formed by the block distortion (or the matching error) is usually multimodal. In most cases, the error surface will contain one or more local minimas. This can be due to a number of reasons like the textured local frame content or the luminance change between frames [14].

Property 3. The distribution of the SAD is spacial directionality, that is to say, the value of the SAD will decrease to the minima in some direction. This is illustrated in Figure 2b, in which the best matching area is around the origin. The value of SAD at other positions falls down to the best matching value in some direction. If the search strategy is the same at every area, a mass of search redundancy is generated. According to the property, the BMA should use the line search in order to quickly determine the best matching area and then employ accurate search strategy to find the best matching point in the area.

2.2. Algorithm development of the LSPS

In the previous section, the basic properties of the block motion fields and the SAD distribution are introduced. The LSPS will use these properties to design the search strategy and the search pattern. The fast BMAs [7, 8, 9, 10, 11, 13] that have been proposed follow a coarse-to-fine approach, which

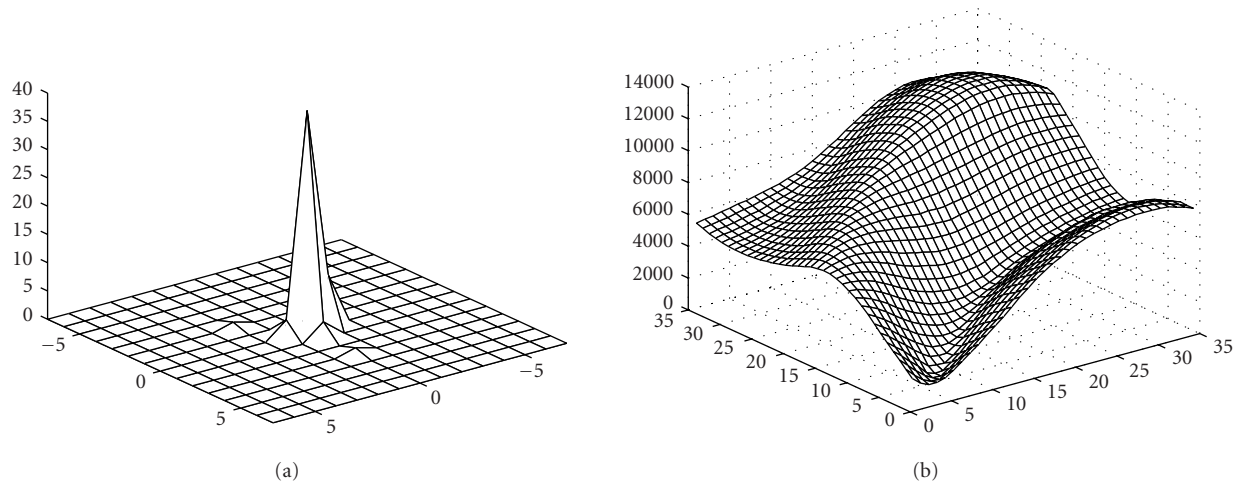


FIGURE 2: The basic properties of the block motion fields and the SAD distribution: (a) center-biased motion vector distribution characteristic and (b) spacial direction SAD distribution characteristic.

is a waste of time for stationary blocks. The LSPS algorithm would also solve the problem by employing the parallel processing idea to realize the coarse orientation and the accurate search in the same step. Figure 3 depicts a basic search-point configuration used in LSPS. Figure 3a is a square pattern (SP), which is used to determine the direction of the *line search* and process the accurate search. The SP includes 17 search points. The outside 8 search points are suppositional points, which are referred to as the “outer pattern.” The other 9 search points form the “inner pattern.” Firstly, the SAD values of the search points on the inner pattern are computed. If the position of the MBD is at the center point, it is the estimated motion vector. If the position of the MBD is not at the center point, the SAD value of the nearest suppositional point on the outer-pattern will be computed. If the value of the suppositional point is smaller than that of the MBD, *line search* is executed in the direction, which is illustrated in Figure 3b. In this step, accurate search and search direction selection are accomplished at the same time, which is the embodiment of the parallel processing idea. When the SAD value of search point in this direction does not decline, the SP is reused to do accurate search and select the new search direction. Figures 3c and 3d show the line search strategy and the positions of the SP, with respect to the previous position, for the next search step along the SP’s face and vertex, respectively.

The above search path strategy using the LSPS algorithm can be summarized as follows.

Starting. The SP (Figure 3a) is placed at (0,0), the center of the search window. The center of the SP is called the original point.

Selecting the search direction. The block distortion measure (BDM) is evaluated for each of the nine candidate search points on the inner pattern. If the minimum BDM point is found at the center of the SP, proceed to *Ending*; Else, the BDM is evaluated for the nearest suppositional point on the outer pattern. If the block distortion value of the sup-

positional point is smaller than that of the minimum BDM point, let the suppositional point be the line search point, proceed to *Line Search*. Else, let the minimum BDM point be the center of the SP, proceed to *Selecting the search direction*.

Line Search. The BDM is evaluated for the next checking point in the direction determined by the original point and line search point, which is illustrated in Figures 3c and 3d. If the SAD value of the checking point is smaller than that of the *line search point*, then let the line search point be the original point and the checking point (like the point 3 in Figure 3c) be the line search point, proceed to *Line Search*; Else, let the line search point be the center of the SP, proceed to *Selecting the search direction*.

Ending. The center point is the estimated motion vector. The current block’s search process is completed.

2.3. Performance analysis of the LSPS

We have explained earlier the algorithm development of LSPS in this paper. This subsection aims to investigate how accurate search and speed improvement could be obtained over the search algorithm. In particular, we are comparing the LSPS algorithm with the DS algorithm.

Table 1 shows the number of the search points of two algorithms for quasi-stationary block (within the 3 pixels) ME. Of course, the computations rely on the assumption that the SAD measure decreases monotonically as the search position moves closer to the optimum position. According to the search strategies of the two algorithms, for quasi-stationary block, the LSPS algorithm could realize the coarse orientation and the accurate search in the same step. However, the DS algorithm has to follow a coarse-to-fine approach. For example, for the stationary blocks, the LSPS algorithm only evaluates 9 search points to locate the optimum position, which evaluates 4 search points less than the DS algorithm. It is clear that, compared with the DS algorithm, the LSPS algorithm is fitter for the small motion blocks.

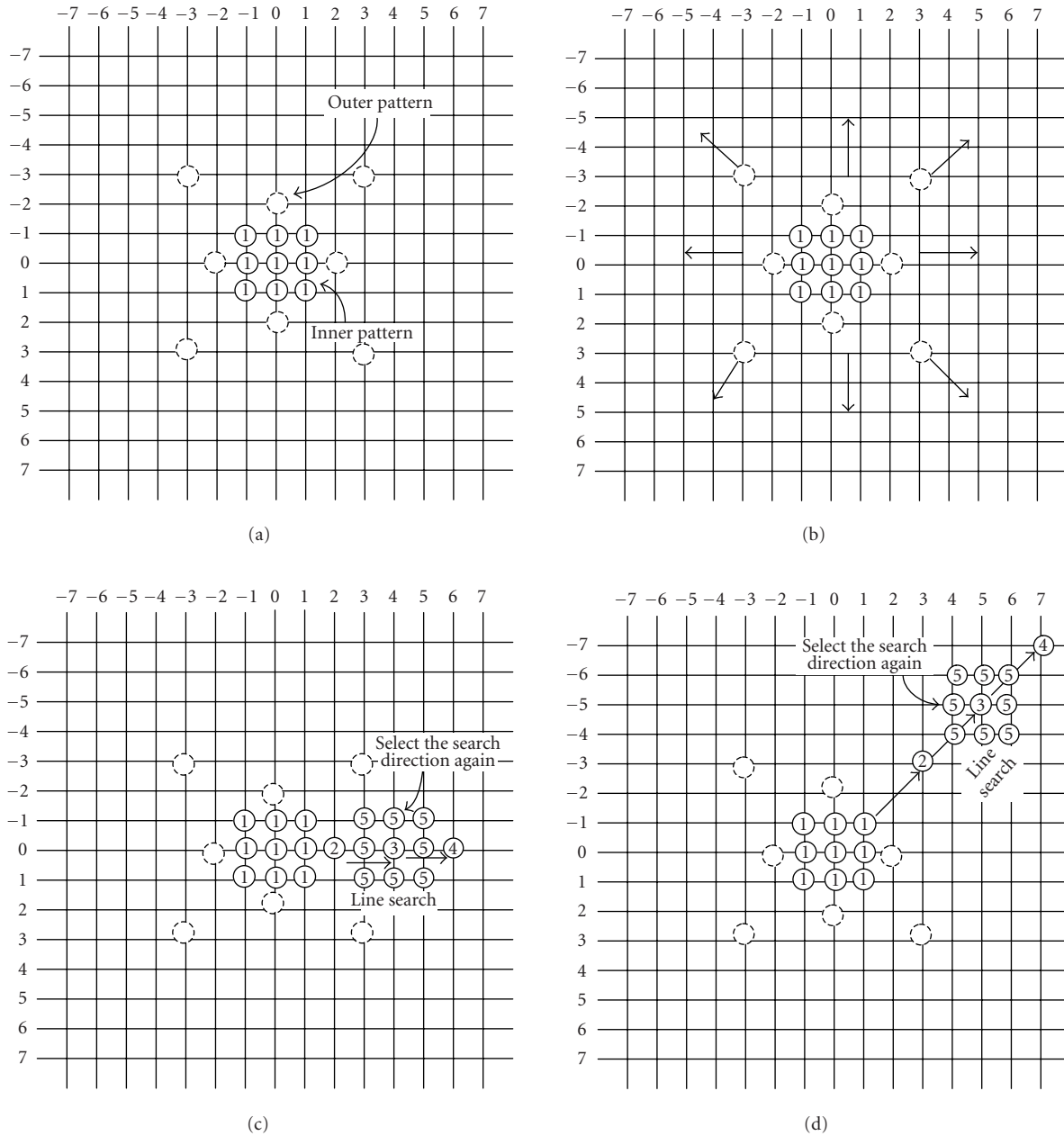


FIGURE 3: The pattern and search strategy of the LSPS algorithm: (a) square pattern (SP), (b) eight directions of the line search, (c) next step along the SP's face, and (d) next step along the SP's vertex.

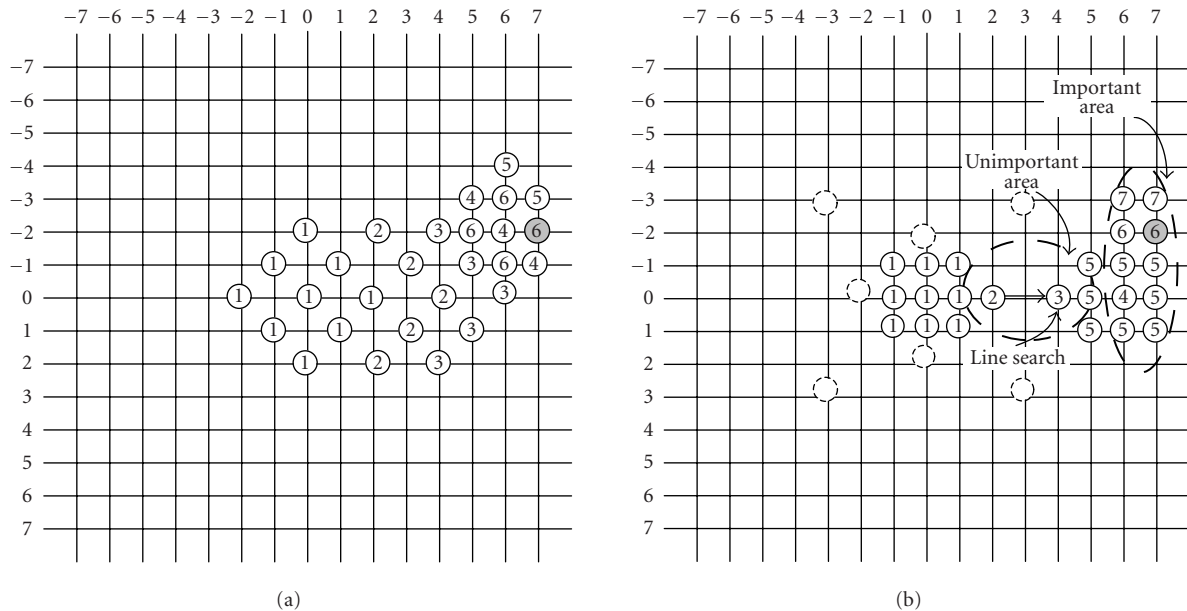
Figure 4 illustrates an example of the search path using the DS algorithm and the LSPS algorithm. To locate a motion vector at (+7, -2), the DS algorithm has performed 28 block evaluations and the LSPS algorithm has only performed 24 block evaluations. The speed improvement of the LSPS algorithm is mainly because the line search is used. The LSPS algorithm performs line search for the unimportant area to reduce the number of the search points. After the important area is arrived at, the square search pattern comprising nine checking points is performed to improve the prediction quality. This example shows that the search speed of the LSPS al-

gorithm is also higher than that of the DS algorithm for large motion blocks.

Table 1 and Figure 4 show that the computational complexity of the LSPS algorithm is lower than that of the DS algorithm for stationary blocks as well as large motion blocks. From the above analysis, we can conclude that the speed improvement of the LSPS algorithm is substantial, which justifies its use over the DS algorithm. The following experimental results would prove that the accuracy of the LSPS algorithm is similar to that of the DS algorithm.

TABLE 1: Comparison between LSPS and DS in terms of the number of the search points for quasi-stationary block.

Algorithm	The position of the best matching point				
	The center	The radius is 1	The radius is $\sqrt{2}$	The radius is 2	The radius is 3
DS	13	13	16	18	18
LSPS	9	12	15	16	18

FIGURE 4: Example of a search path strategy using DS and LSPS to locate a motion vector at $(+7, -2)$. (a) The search path using DS (28 search points). (b) The search path using LSPS (23 search points).

3. SIMULATION RESULTS AND DISCUSSIONS

In this section, we are going to give actual experimental results on the performance of the proposed LSPS from the viewpoint of the computational complexity as well as the accuracy of the estimated vectors. In all of our simulations, block size $N = 16$, search window size $W = 7$, and the SAD-BDM were used. The SAD is given by

$$\text{SAD}(i, j) = \sum_{m=1}^M \sum_{n=1}^N |f_k(m, n) - f_{k-1}(m+i, n+j)|. \quad (1)$$

In our simulations, four various types of test sequences were used: the “Claire,” the “Flower Garden,” the “Foreman,” and the “Football” sequences. The “Claire” sequence is a typical videoconferencing scene with limited object motion and a stationary background. The Flower Garden sequence consists mainly of stationary objects, but with a fast camera panning motion. The Foreman sequence has large local object motions. Finally, the Football sequence consists of complex motions and the movement of the motions is big. In each of the four test sequences, the first 100 frames were used. We compared the LSPS against four other BMAs—FS, TSS, FSS, and DS.

For BMA, computational complexity could be measured by average number of search points required for each motion vector’s estimation. Table 2 shows the average number of search points and the speedup ratio of each algorithm with reference to the FS algorithm. Table 3, on the other hand, compares the prediction quality of the simulated BMA algorithms in terms of average luma PSNR in decibels. As expected, the FS algorithm provides the best prediction quality, but at the expense of a very high computational complexity. From the result of Table 2, we can find that the LSPS algorithm provides the highest speedup ratios. For the image sequence with small motion content, such as “talking-head” sequences (e.g., Claire), it is remarkable that the LSPS algorithm provides the speedup ratio up to 23.41. A reasonable explanation for this is that the parallel processing idea is introduced in the LSPS algorithm. From the result of Table 3, it can be concluded that the LSPS algorithm provide the better prediction quality for the image sequence with large motion content (e.g., Foreman). The above observations are further emphasized in Figure 5, which shows the actual performance for 100 frames of Football sequence.

4. CONCLUSION AND FUTURE RESEARCH

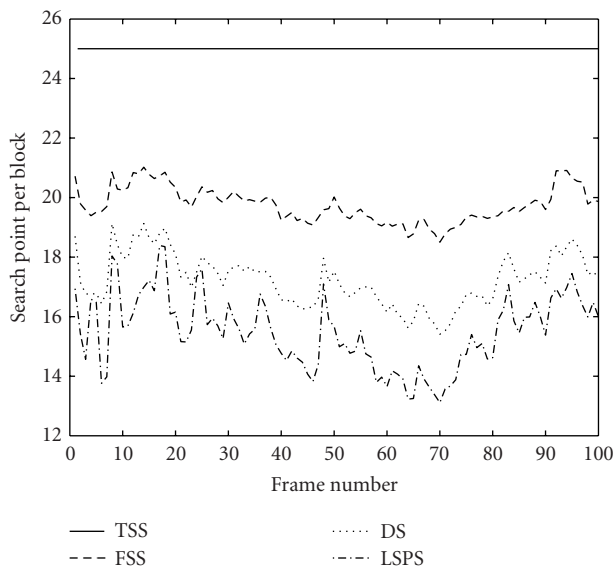
In this paper, the basic properties of the block motion fields and the SAD distribution are analyzed. Based on the analysis

TABLE 2: Comparison between different BMAs in terms of the average search points per block.

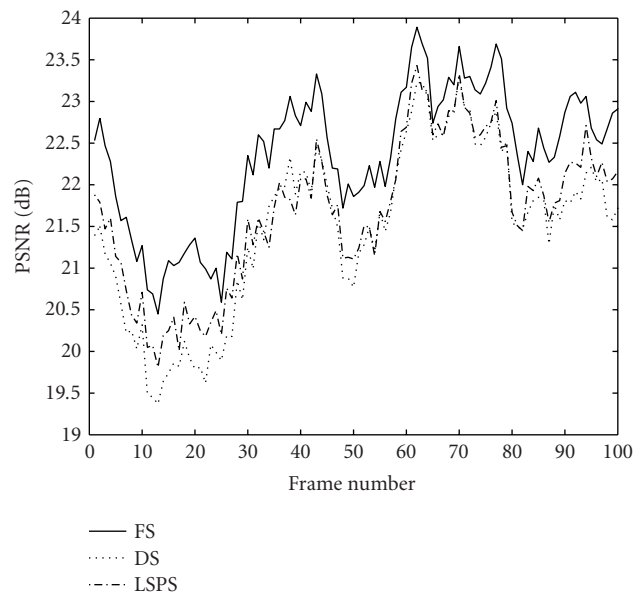
Algorithm	Claire		Flower Garden		Foreman		Football	
	Avg. search points	Speedup	Avg. search points	Speedup	Avg. search points	Speedup	Avg. Search Points	Speedup
FS	225	1.00	225	1.00	225	1.00	225	1.00
TSS	25	9.00	25	9.00	25	9.00	25	9.00
FSS	16.56	13.59	19.06	11.80	19.36	11.62	19.39	11.60
DS	13.38	16.82	17.01	13.23	16.56	13.58	16.84	13.36
LSPS	9.61	23.41	15.06	14.94	15.03	14.97	14.99	15.01

TABLE 3: Comparison between different BMAs in terms of prediction quality.

Algorithm	Claire		Flower Garden		Foreman		Football	
	PSNR	Δ PSNR	PSNR	Δ PSNR	PSNR	Δ PSNR	PSNR	Δ PSNR
FS	41.46	0.00	25.64	0.00	34.71	0.00	22.56	0.00
TSS	41.14	-0.32	23.41	-2.23	32.19	-2.52	21.53	-1.03
FSS	41.40	-0.06	25.02	-0.62	33.81	-0.90	21.77	-0.79
DS	41.45	-0.01	25.30	-0.34	34.11	-0.60	21.74	-0.83
LSPS	41.46	0.00	25.35	-0.29	34.38	-0.33	21.81	-0.75



(a)



(b)

FIGURE 5: Performance comparison between different BMAs when applied to 100 frames of the Football sequence: (a) computational complexity and (b) prediction quality.

and the parallel processing idea, a new LSPS algorithm for fast block matching ME is developed. Like DS, the proposed techniques can be applied to any block-based compression standard and GOP structure, and the search window size is not restricted by the searching strategy in the LSPS algorithm. Experimental results show that, compared with the DS algorithm, the LSPS algorithm achieves close prediction quality as well as greatly reduces its computational complexity.

There are still several areas of work to be conducted.

- (1) The predictive quality of the LSPS algorithm is not as good as that of the FS algorithm. The problem can be tackled by adopting the multiple candidates. For this, we have done some preliminary researches.
- (2) There is a high correlation between the motion vectors of adjacent blocks, which can be used to improve the LSPS algorithm's effectiveness. This would be investigated in our following work.

ACKNOWLEDGMENT

This work was supported by the National Science Foundation of China (Grant 69975015).

REFERENCES

- [1] ISO/IEC CD 11172-2 (MPEG-1 Video), *Information technology—Coding of moving pictures and associated audio for digital storage media at up to about 1.5 Mb/s: Video*, 1993.
- [2] ISO/IEC CD 13818-2 (ITU-T H.262, MPEG-2 Video), *Information technology—Generic coding of moving pictures and associated audio information: Video—Part 2*, 2000.
- [3] ISO/IEC CD 14496-2 (MPEG-4), *Information technology—Coding of audio-visual objects—Part 2: Video*, 2001.
- [4] CCITT SGXV, *Video Codec for Audiovisual Services at p*64 kbit/s Recommendation H.261*, Geneva, Switzerland, 1993.
- [5] International Telecommunication Union, *Video Coding for Low Bitrate Communication*, ITU-T Recommendation H.263, 1996.
- [6] ISO/IEC 11496-10, *Advanced Video Coding. ITU-T Recommendation H.264, Final Committee Draft, Document JVT-E022*, September 2002.
- [7] R. Li, B. Zeng, and M. L. Liou, "A new three-step search algorithm for block motion estimation," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 4, no. 4, pp. 438–442, 1994.
- [8] L.-M. Po and W.-C. Ma, "A novel four-step search algorithm for fast block motion estimation," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 6, no. 3, pp. 313–317, 1996.
- [9] J. Jain and A. Jain, "Displacement measurement and its application in interframe image coding," *IEEE Trans. Communications*, vol. 29, no. 12, pp. 1799–1808, 1981.
- [10] M. Ghanbari, "The cross-search algorithm for motion estimation [image coding]," *IEEE Trans. Communications*, vol. 38, no. 7, pp. 950–953, 1990.
- [11] S. Zhu and K.-K. Ma, "A new diamond search algorithm for fast block-matching motion estimation," *IEEE Trans. Image Processing*, vol. 9, no. 2, pp. 287–290, 2000.
- [12] "Special issue on MPEG-4," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 7, no. 1, 1997.
- [13] J. Y. Tham, S. Ranganath, M. Ranganath, and A. A. Kassim, "A novel unrestricted center-biased diamond search algorithm for block motion estimation," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 8, no. 4, pp. 369–377, 1998.
- [14] M. E. Al-Mualla, C. N. Canagarajah, and D. R. Bull, "Simplex minimization for single- and multiple-reference motion estimation," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 11, no. 12, pp. 1209–1220, 2001.

Guo Bao-long was born in 1962. He received the B.S. degree, the M.S. degree, and the Ph.D. degree from Xidian University in 1984, 1988, and 1995, respectively, all in communication and electronic system. He is currently a Full Professor of pattern recognition and intelligent system at Xidian University. His research interests are neural networks, pattern recognition, image processing, and image communication.



Ding Gui-guang was born in 1976. He received the B.S. degree in measurement control engineering and apparatus, the M.S. degree in information and communication engineering, and the Ph.D. degree in circuit and system from Xidian University in 1999, 2002, and 2004, respectively. He is currently a postdoctor with the Broadband Networks & Digital Media Lab of Automation Department, Tsinghua University. His research interests include image and video coding, image processing, and multimedia communication.

